PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-318766
(43)Date of publication of application : 16.11.2001

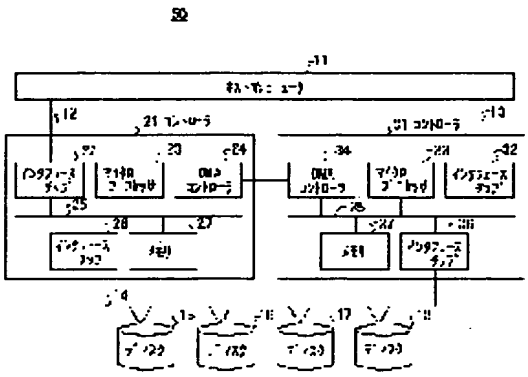
(51)Int.Cl. G06F 3/06
G06F 12/08
G06F 12/16
G11B 19/02

(21)Application number : 2000-135013
(22)Date of filing : 08.05.2000

(71)Applicant : NEC SOFTWARE SHIKOKU LTD
(72)Inventor : SHIRAISHI KAZUYA

(54) DISK ARRAY DEVICE AND CACHE MEMORY CONTROL METHOD

(57)Abstract:
PROBLEM TO BE SOLVED: To provide a disk array device and a cache memory control method for executing second and succeeding pieces of write command processing without putting loads on the processor of a main controller.
SOLUTION: This disk array device is composed of two controllers and executes the double write of write data in the two controllers without using the hardware of a shared memory. The write data are held in both of the two controllers and a means is provided for making the write data held in the other one of the two controllers mutually referable when a fault is generated in one of the controllers.



LEGAL STATUS

[Date of request for examination] 17.04.2001
[Date of sending the examiner's decision of rejection] 12.02.2003
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

No. 1

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2001－318766
（P2001－318766A）
(43)公開日　平成13年11月16日(2001.11.16)

(51)Int.Cl.'	識別記号	FI		テーマコード'（参考）
G06F 3/06	304	G06F 3/06	304E	5B005
	540		540	5B018
12/08	511	12/08	511Z	5B065
	541		541C	5D066
	557		557

審査請求　有　請求項の数14　ＯＬ　（全 12 頁）　続頁に続く

(21)出願番号　特願2000－135013(P2000－135013)

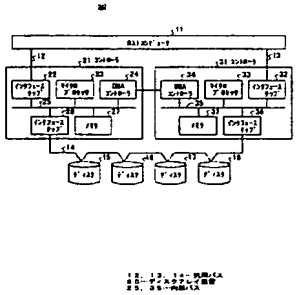
(22)出願日　平成12年５月８日(2000.5.8)

(71)出願人　000180379
四国日本電気ソフトウェア株式会社
愛媛県松山市衣山４丁目760番地
(72)発明者　白石　和也
愛媛県松山市衣山４丁目760　四国日本電気ソフトウエア株式会社内
(74)代理人　100097113
弁理士　堀　城之

Ｆターム(参考)　5B005 JJ01 JJ11 MM11 NN02 PP04
5B018 GA04 HA02 HA35 MA03 MA14
5B065 BA01 CA12 CA30 CC08 CE12
CE14 EA12 EA31
5D066 BA02 BA08

(54)【発明の名称】　ディスクアレイ装置およびキャッシュメモリ制御方法

(57)【要約】
【課題】　本発明は、２回目以降のライトコマンド処理においてメイトコントローラのプロセッサに負荷をかけることなく実行できるようになるディスクアレイ装置およびキャッシュメモリ制御方法を提供することを課題とする。
【解決手段】　２つのコントローラで構成されかつ共有メモリのハードウェアを用いることなく当該２つのコントローラでライトデータの２重書きを実行するディスクアレイ装置であって、ライトデータを前記２つのコントローラの両方に保持するとともに、前記コントローラの一方に故障が発生した際に前記２つのコントローラの他方に保持されているライトデータを相互参照可能とする手段を有する。



【特許請求の範囲】
【請求項１】　２つのコントローラで構成されかつ共有メモリのハードウェアを用いることなく当該２つのコントローラでライトデータの２重書きを実行するディスクアレイ装置であって、
ライトデータを前記２つのコントローラの両方に保持するとともに、前記コントローラの一方に故障が発生した際に前記２つのコントローラの他方に保持されているライトデータを相互参照可能とする手段を有することを特徴とするディスクアレイ装置。
【請求項２】　ホストコンピュータとディスクとの間に前記２つのコントローラの両方を有し、
前記ホストコンピュータと前記２つのコントローラの両方が汎用バスで接続されるとともに、前記２つのコントローラの両方と前記ディスクが汎用バスで接続されていることを特徴とする請求項１に記載のディスクアレイ装置。
【請求項３】　前記２つのコントローラのそれぞれは、マイクロプロセッサと、リードまたはライトのデータをキャッシングしかつプロセッサの制御情報を記憶するメモリと、前記ホストコンピュータとの通信を制御する第１のインタフェースチップと、前記ディスクとの通信を制御する第２のインタフェースチップと、前記メモリの任意アドレスのデータを前記２つのコントローラの他方の前記メモリの任意アドレスに送信できるＤＭＡコントローラを有していることを特徴とする請求項１または２に記載のディスクアレイ装置。
【請求項４】　セグメント単位でキャッシュを管理する前記メモリには、セグメント管理情報と、ビットマップと、ライトキャッシュが定義されていることを特徴とする請求項３に記載のディスクアレイ装置。
【請求項５】　前記セグメント管理情報は各セグメントの状態を保持するものであって、シーズの状態を示すシーズフラグと、ロジカルユニット番号と、ロジカルブロックアドレスと、自コントローラの両方のビットマップ格納位置を示す自ビットマップアドレスと、自コントローラのキャッシュ位置を示す自キャッシュアドレスと、前記２つのコントローラの他方のビットマップ格納位置を示す他ビットマップアドレスと、前記２つのコントローラの他方のキャッシュ位置を示す他キャッシュアドレスを有することを特徴とする請求項４に記載のディスクアレイ装置。
【請求項６】　前記ビットマップは、各セグメントに対応するキャッシュのどの位置に有効なデータが存在するかを示すマップであることを特徴とする請求項５に記載のディスクアレイ装置。
【請求項７】　前記ライトキャッシュは、各セグメントに対応するキャッシュメモリであることを特徴とする請求項６に記載のディスクアレイ装置。
【請求項８】　前記ホストコンピュータからライトコマンドを受信した際に前記コントローラの一方は、自己の前記セグメント管理情報を参照して自身がシーズしているセグメントにヒットしているかを確認し、ヒットしていなければ汎用バスを介して前記２つのコントローラの他方にシーズコマンドを送信してセグメントをシーズし、前記ホストコンピュータからライトデータを受信して自己の前記ライトキャッシュに格納し当該格納位置に対応した自己の前記ビットマップを有効状態に更新し、自己の前記ライトキャッシュのデータおよび自己の前記ビットマップを前記２つのコントローラの他方の前記ＤＭＡコントローラを介して前記２つのコントローラの他方側の前記ライトキャッシュおよび前記ビットマップに送信してライトコマンドを受信し、前記コントローラの一方がシーズしているセグメントにヒットした場合に前記２つのコントローラの他方にシーズコマンドを送信することなく、前記ホストコンピュータからライトデータを受信して自己の前記ライトキャッシュに格納し当該格納位置に対応した自己の前記ビットマップを有効状態に更新し、自己の前記ライトキャッシュのデータおよび自己の前記ビットマップを前記２つのコントローラの他方の前記ＤＭＡコントローラを介して前記２つのコントローラの他方側の前記ライトキャッシュおよび前記ビットマップに送信することを特徴とする請求項７に記載のディスクアレイ装置。
【請求項９】　２つのコントローラで構成されかつ共有メモリのハードウェアを用いることなく当該２つのコントローラでライトデータの２重書きを実行するディスクアレイ装置に対して、ライトデータを前記２つのコントローラの両方に保持するとともに、前記コントローラの一方に故障が発生した際に前記２つのコントローラの他方に保持されているライトデータを相互参照可能とする工程を有することを特徴とするキャッシュメモリ制御方法。
【請求項１０】　前記２つのコントローラのそれぞれに設けられリードまたはライトのデータをセグメント単位でキャッシングしかつプロセッサの制御情報を記憶するメモリに、セグメント管理情報と、ビットマップと、ライトキャッシュを定義する工程を有することを特徴とする請求項９に記載のキャッシュメモリ制御方法。
【請求項１１】　前記セグメント管理情報は各セグメントの状態を保持するものであって、シーズの状態を示すシーズフラグと、ロジカルユニット番号と、ロジカルブロックアドレスと、自コントローラの両方のビットマップ格納位置を示す自ビットマップアドレスと、自コントローラのキャッシュ位置を示す自キャッシュアドレスと、前記２つのコントローラの他方のビットマップ格納位置を示す他ビットマップアドレスと、前記２つのコントローラの他方のキャッシュ位置を示す他キャッシュアドレスを有することを特徴とする請求項１０に記載のキャッシュメモリ制御方法。

【請求項１２】　前記ビットマップは、各セグメントに対応するキャッシュのどの位置に有効なデータが存在するかを示すマップであることを特徴とする請求項１１に記載のキャッシュメモリ制御方法。
【請求項１３】　前記ライトキャッシュは、各セグメントに対応するキャッシュメモリであることを特徴とする請求項１２に記載のキャッシュメモリ制御方法。
【請求項１４】　前記ホストコンピュータからライトコマンドを受信した際に前記コントローラの一方は、自己の前記セグメント管理情報を参照して自身がシーズしているセグメントにヒットしているかを確認する工程と、ヒットしていなければ汎用バスを介して前記２つのコントローラの他方にシーズコマンドを送信してセグメントをシーズする工程と、前記ホストコンピュータからライトデータを受信して自己の前記ライトキャッシュに格納し当該格納位置に対応した自己の前記ビットマップを有効状態に更新する工程と、自己の前記ライトキャッシュのデータおよび自己の前記ビットマップを前記２つのコントローラの他方側の前記ライトキャッシュおよび前記ビットマップに送信してライトコマンドを受信する工程と、前記コントローラの一方がシーズしているセグメントにヒットした場合に前記２つのコントローラの他方にシーズコマンドを送信することなく、前記ホストコンピュータからライトデータを受信して自己の前記ライトキャッシュに格納し当該格納位置に対応した自己の前記ビットマップを有効状態に更新する工程と、自己の前記ライトキャッシュのデータおよび自己の前記ビットマップを前記２つのコントローラの他方の前記ＤＭＡコントローラを介して前記２つのコントローラの他方側の前記ライトキャッシュおよび前記ビットマップに送信することを特徴とする請求項１３に記載のキャッシュメモリ制御方法。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、２コントローラ構成でかつ共有メモリのハードウェアを持たないディスクアレイ装置のキャッシュメモリ制御技術に係り、特にメイトコントローラのセグメントをシーズしてそのキャッシュアドレスとビットマップを保持することにより、２回目以降のライトコマンド処理においてメイトコントローラのプロセッサに負荷をかけることなく実行できるようになるディスクアレイ装置およびキャッシュメモリ制御方法に関する。
【０００２】
【従来の技術】従来、ライトデータの２重書き技術としては、例えば、キャッシュメモリを２つのコントローラから独立させて設け当該両コントローラからアクセス可能な共有メモリ構成とする方式と、コントローラ内にキャッシュメモリを配置し何らかの方法でキャッシュデータをメイトコントローラに送信する方式等が開示されている。
【０００３】このような先行技術としては、例えば、特開平１０－２２２４２３号公報（第１従来技術）に記載のキャッシュメモリ制御方式がある。図５は、第１従来技術のキャッシュメモリ制御方式を説明するための機能ブロック図である。図５を参照すると、第１従来技術のキャッシュメモリ制御方式は、プロセッサ１００、プロセッサ２００、共有メモリ３００、および入出力制御装置４００が共有バス１０に接続されたＭＰシステムである。
【０００４】プロセッサ１００は、ライトバック方式で制御されるキャッシュメモリ１１０と、キャッシュメモリ１１０に入出力されるアドレス信号線、データ信号線、制御信号線などを制御するキャッシュメモリ制御回路１２０と、キャッシュメモリ１１０のアドレス部と状態部の写しを格納するコピータグ１３０と、コピータグ１３０に入出力されるアドレス信号線、データ信号線、制御信号線などを制御するコピータグ制御回路１４０と、演算処理やプログラムの実行制御を行う演算制御回路１５０と、演算制御回路１５０と共有バス１０とのインターフェースを制御するバスインターフェース回路１６０と、共有バス１０上のアクセス要求を検出しコピータグ制御回路１４０やバスインターフェース回路１６０への指示を行うバス監視回路１７０とで構成される。キャッシュメモリ１１０は、格納している各ブロックの上位アドレスを記憶するアドレス部１１１と、各ブロックの状態値を格納する状態部１１２と、各ブロックのデータを記憶するデータ部１１３とから構成される。ここでいう上位アドレスとは、アドレスをビット列としてみた場合に、ブロック内のバイト位置を指定するビット列およびキャッシュメモリ内のブロック位置を指定するビット列を除いた上位ビット列を指す。アドレス部１１１と状態部１１２とを合わせて、タグ部と呼ぶ。コピータグ１３０は、キャッシュメモリ１１０のタグ部の写しを格納するものであり、キャッシュメモリ１１０のアドレス部１１１の写しを格納するアドレス部１３１と、キャッシュメモリ１１０の状態部１１２の写しを格納する状態部１３２とから構成される。キャッシュメモリ１１０の状態部１１２とコピータグ１３０の状態部１３２とはそれぞれ、ブロックごとに４つの状態値を格納することができる。４つの状態値とは、無効状態、共有状態、排他一致状態、および排他変更状態である。キャッシュメモリ制御回路１２０は、キャッシュメモリ１１０の任意のブロックの排他変更状態への遷移とそのブロック位置をコピータグ１３０に通知する信号を信号線１２１を通して出力する。
【０００５】プロセッサ２００は、プロセッサ１００と同様に構成されており、プロセッサ２００の構成要素もそれぞれプロセッサ１００の相当する構成要素と同様である。すなわち、プロセッサ２００は、アドレス部２１１、状態部２１２、およびデータ部２１３を備えるキャッシュメモリ２１０と、キャッシュメモリ制御回路２２０と、アドレス部２３１および状態部２３２を備えるコピータグ２３０と、コピータグ制御回路２４０と、演算制御回路２５０と、バスインターフェース回路２６０と、バス監視回路２７０とで構成される。
【０００６】共有メモリ３００は、共有バス１０に接続され、プロセッサ１００、プロセッサ２００、および入出力制御装置４００から共有バス１０上に要求されたメモリリードアクセスやメモリライトアクセスに対して応答可能な記憶装置である。
【０００７】入出力制御装置４００は、特定の入出力装置、例えば磁気ディスク装置（図示せず）とのインターフェースを持ち、共有バス１０を介して共有メモリ３００にリード／ライト可能な装置である。
【０００８】共有バス１０は、接続された各装置が相互にアクセス可能なためのアドレス信号線、データ信号線、その他リードライトコマンド信号線、バス調停信号線、タイミング信号線などの制御信号線を含むバスである。共有バス１０は、特に、リード要求に対する応答データがプロセッサにより変更されているか否かを通知する変更信号線５１１と、リード要求に対する応答データが複数のプロセッサで共有されるか否かを通知する共有信号線５１２とを含む。
【０００９】また他の従来技術としては、例えば、特開平６－６７９７９号公報（第２従来技術）に記載の主記憶装置の制御方式がある。図６は、第２従来技術の主記憶装置の制御方式を説明するための機能ブロック図である。
【００１０】図６を参照すると、第２従来技術の主記憶装置は、現用系情報処理システム（以下ＳＹＳＡ）６１と、障害時に処理を引き継ぐ待機情報処理システム（ＳＹＳＢ）６１２とから成っている。ＳＹＳＡ６１１はプロセッサ（以下ＣＰＵ）１ａと、ＣＰＵ１ａにアドレスバス５ａ、データバス６ａで接続されたコピーバック・キャッシュメモリ（以下ＣＢＣＭ）２ａと、ＣＢＣＭ２ａにアドレスバス７ａ、データバス８ａで接続された主記憶装置（以下ＭＥＭ）３ａと、ＣＰＵ１ａにアドレスバス５ａ、データバス６ａで接続されたコピーバック・バッファ装置（以下ＣＢＢＭ）４ａとを有して構成している。
【００１１】また、ＳＹＳＢ６１２はプロセッサ（以下ＣＰＵ）１ｂと、ＣＰＵ１ｂにアドレスバス５ｂ、データバス６ｂで接続されたコピーバック・キャッシュメモリ（以下ＣＢＣＭ）２ｂと、ＣＢＣＭ２ｂにアドレスバス７ｂ、データバス８ｂで接続された主記憶装置（以下ＭＥＭ）３ｂと、ＣＰＵ１ｂにアドレスバス５ｂ、データバス６ｂで接続されたコピーバック・バッファ装置（以下ＣＢＢＭ）４ｂとを有して構成している。
【００１２】ＳＹＳＡ６１１のＣＢＢＭ４ａはＳＹＳＢ６１２のＭＥＭ３ｂにアドレスバス９ａとデータバス１０ａとで接続され、ＳＹＳＢ６１２のＣＢＢＭ４ｂはＳＹＳＡ６１１のＭＥＭ３ａにアドレスバス９ｂとデータバス１０ｂとで接続され、本適用例はＳＹＳＡ６１１とＳＹＳＢ６１２とが互いに対称形に接続されたシステム構成となっている。
【００１３】このように、第２従来技術では、コントローラ内にキャッシュメモリを配置し、ハードウェアによってメイトコントローラにライトデータを送信する仕組みとして、ＣＰＵからコピーバック・キャッシュメモリ（ＣＢＣＭ）に送信されるライトデータをＣＢＢＭで横取りしてメイトコントローラに送信しているが、例えば、ＰＣＩバス（Ｐｅｒｉｐｈｅｒａｌ Ｃｏｍｐｏｎｅｎｔ Ｉｎｔｅｒｃｏｎｎｅｃｔ ｂｕｓ：システムバス・ローカルバス）のような汎用バスを用いる場合においては横取りという仕様は定義されていないため、内部バスはオリジナルバスで構成することになる。
【００１４】また第３の従来技術として、例えば、図７に示すようなディスクアレイ装置が開示されている。図７は、第３従来技術のディスクアレイ装置を説明するための機能ブロック図である。すなわち、第３従来技術のディスクアレイ装置は、ホストコンピュータ１１とディスク１５、…、１８との間に２つのコントローラ２１、３１を有するものである。ホストコンピュータ１１とコントローラ２１、３１は、例えば、ＳＣＳＩ（スカジー：スモール・コンピュータ・システム・インタフェース）バスのような汎用バス１２、１３で接続されている。コントローラ２１、３１とディスク１５、…、１８は、例えば、ＳＣＳＩバスのような汎用バス１４で接続されている。
【００１５】コントローラ２１は、マイクロプロセッサ２３と、リードまたはライトのデータをキャッシングしかつプロセッサの制御情報を記憶するメモリ２７と、ホストコンピュータ１１との通信を制御するインタフェースチップ２２と、ディスク１５、…、１８との通信を制御するインタフェースチップ２６を有し、コントローラ内の各チップは内部バス２５で接続されている。
【００１６】同様に、コントローラ３１は、マイクロプロセッサ３３と、リードまたはライトのデータをキャッシングしかつプロセッサの制御情報を記憶するメモリ３７と、ホストコンピュータ１１との通信を制御するインタフェースチップ３２と、ディスク１５、…、１８との通信を制御するインタフェースチップ３６を有し、コントローラ内の各チップは内部バス３５で接続されている。
【００１７】
【発明が解決しようとする課題】しかしながら、上記第１従来技術には、共有メモリ３００は、コントローラの故障交換によって一緒に取り外されないように、独立し

た基板（パッケージ）で作成されているため、当該メモリの故障を考慮すると、共有メモリ３００の２重化が必要になり、回路構成が大きくコストが高くなるという欠点があり、中小型システムでは採用することが難しいという問題点があった。

【００１８】また、上記第２従来技術は、内部バスをオリジナルバスで構成することになるため、汎用バスを採用できる場合はインタフェースチップなどを汎用品の中から自由に選択でき高性能な装置を安価に入手できるメリットがあるものの、オリジナルバスではこのような構成が難しいという問題点があった。

【００１９】そして、上記第３従来技術は、２重書きにはディスク側の汎用バス１４を使用するため、処理のオーバーヘッドが大きいという問題点があり、また、メイトコントローラのプロセッサに負荷がかかるという問題点もあった。

【００２０】本発明は斯かる問題点を鑑みてなされたものであり、その目的とするところは、メイトコントローラのセグメントをシーズしてそのキャッシュアドレスとビットマップを保持することにより、２回目以降のライトコマンド処理においてメイトコントローラのプロセッサに負荷をかけることなく実行できるようになるディスクアレイ装置およびキャッシュメモリ制御方法を提供する点にある。

【００２１】

【課題を解決するための手段】この発明の請求項１に記載の発明の要旨は、２つのコントローラで構成されかつ共有メモリのハードウェアを用いることなく当該２つのコントローラでライトデータの２重書きを実行するディスクアレイ装置であって、ライトデータを前記２つのコントローラの両方に保持するとともに、前記コントローラの一方に故障が発生した際に前記２つのコントローラの他方に保持されているライトデータを相互参照可能とする手段を有することを特徴とするディスクアレイ装置に存する。また、この発明の請求項２に記載の発明の要旨は、ホストコンピュータとディスクとの間に前記２つのコントローラの両方を有し、前記ホストコンピュータと前記２つのコントローラの両方が汎用バスで接続されるとともに、前記２つのコントローラの両方と前記ディスクが汎用バスで接続されていることを特徴とする請求項１に記載のディスクアレイ装置に存する。また、この発明の請求項３に記載の発明の要旨は、前記２つのコントローラのそれぞれは、マイクロプロセッサと、リードまたはライトのデータをキャッシングしかつプロセッサの制御情報を記憶するメモリと、前記ホストコンピュータとの通信を制御する第１のインタフェースチップと、前記ディスクとの通信を制御する第２のインタフェースチップと、前記メモリの任意アドレスのデータを前記２つのコントローラの他方の前記メモリの任意アドレスに送信できるＤＭＡコントローラを有していることを特徴とする請求項１または２に記載のディスクアレイ装置に存する。また、この発明の請求項４に記載の発明の要旨は、セグメント単位でキャッシュを管理する前記メモリには、セグメント管理情報と、ビットマップと、ライトキャッシュが定義されていることを特徴とする請求項３に記載のディスクアレイ装置に存する。また、この発明の請求項５に記載の発明の要旨は、前記セグメント管理情報は各セグメントの状態を保持するものであって、シーズの状態を示すシーズフラグと、ロジカルユニット番号と、ロジカルブロックアドレスと、自コントローラの両方のビットマップ格納位置を示す自ビットマップアドレスと、自コントローラのキャッシュ位置を示す自キャッシュアドレスと、前記２つのコントローラの他方のビットマップ格納位置を示す他ビットマップアドレスと、前記２つのコントローラの他方のキャッシュ位置を示す他キャッシュアドレスを有することを特徴とする請求項４に記載のディスクアレイ装置に存する。また、この発明の請求項６に記載の発明の要旨は、前記ビットマップは、各セグメントに対応するキャッシュのどの位置に有効なデータが存在するかを示すマップであることを特徴とする請求項５に記載のディスクアレイ装置に存する。また、この発明の請求項７に記載の発明の要旨は、前記ライトキャッシュは、各セグメントに対応するキャッシュメモリであることを特徴とする請求項６に記載のディスクアレイ装置に存する。また、この発明の請求項８に記載の発明の要旨は、前記ホストコンピュータからライトコマンドを受信した際に前記コントローラの一方は、自己の前記セグメント管理情報を参照して自身がシーズしているセグメントにヒットしているかを確認し、ヒットしていなければ汎用バスを介して前記２つのコントローラの他方にシーズコマンドを送信してセグメントをシーズし、前記ホストコンピュータからライトデータを受信して自己の前記ライトキャッシュに格納し当該格納位置に対応した自己の前記ビットマップを有効状態に更新し、自己の前記ライトキャッシュのデータおよび自己の前記ビットマップを前記２つのコントローラの他方の前記ＤＭＡコントローラを介して前記２つのコントローラの他方側の前記ライトキャッシュおよび前記ビットマップに送信してライトコマンドを受信し、前記コントローラの一方がシーズしているセグメントにヒットした場合に前記２つのコントローラの他方にシーズコマンドを送信することなく、前記ホストコンピュータからライトデータを受信して自己の前記ライトキャッシュに格納し当該格納位置に対応した自己の前記ビットマップを有効状態に更新し、自己の前記ライトキャッシュのデータおよび自己の前記ビットマップを前記２つのコントローラの他方の前記ＤＭＡコントローラを介して前記２つのコントローラの他方の前記ライトキャッシュと前記ビットマップに送信することを特徴とする請求項７に記載のディスクアレイ装置に存する。また、この発明の請求項９に

記載の発明の要旨は、２つのコントローラで構成されかつ共有メモリのハードウェアを用いることなく当該２つのコントローラでライトデータの２重書きを実行するディスクアレイ装置に対して、ライトデータを前記２つのコントローラの両方に保持するとともに、前記コントローラの一方に故障が発生した際に前記２つのコントローラの他方に保持されているライトデータを相互参照可能とする工程を有することを特徴とするキャッシュメモリ制御方法に存する。また、この発明の請求項１０に記載の発明の要旨は、前記２つのコントローラのそれぞれに設けられリードまたはライトのデータをセグメント単位でキャッシングしかつプロセッサの制御情報を記憶するメモリに、セグメント管理情報と、ビットマップと、ライトキャッシュを定義する工程を有することを特徴とする請求項９に記載のキャッシュメモリ制御方法に存する。また、この発明の請求項１１に記載の発明の要旨は、前記セグメント管理情報は各セグメントの状態を保持するものであって、シーズの状態を示すシーズフラグと、ロジカルユニット番号と、ロジカルブロックアドレスと、自コントローラの両方のビットマップ格納位置を示す自ビットマップアドレスと、自コントローラのキャッシュ位置を示す自キャッシュアドレスと、前記２つのコントローラの他方のビットマップ格納位置を示す他ビットマップアドレスと、前記２つのコントローラの他方のキャッシュ位置を示す他キャッシュアドレスを有することを特徴とする請求項１０に記載のキャッシュメモリ制御方法に存する。また、この発明の請求項１２に記載の発明の要旨は、前記ビットマップは、各セグメントに対応するキャッシュのどの位置に有効なデータが存在するかを示すマップであることを特徴とする請求項１１に記載のキャッシュメモリ制御方法に存する。また、この発明の請求項１３に記載の発明の要旨は、前記ライトキャッシュは、各セグメントに対応するキャッシュメモリであることを特徴とする請求項１２に記載のキャッシュメモリ制御方法に存する。また、この発明の請求項１４に記載の発明の要旨は、前記ホストコンピュータからライトコマンドを受信した際に前記コントローラの一方は、自己の前記セグメント管理情報を参照して自身がシーズしているセグメントにヒットしているかを確認する工程と、ヒットしていなければ汎用バスを介して前記２つのコントローラの他方にシーズコマンドを送信してセグメントをシーズする工程と、前記ホストコンピュータからライトデータを受信して自己の前記ライトキャッシュに格納し当該格納位置に対応した自己の前記ビットマップを有効状態に更新する工程と、自己の前記ライトキャッシュのデータおよび自己の前記ビットマップを前記２つのコントローラの他方のＤＭＡコントローラを介して前記２つのコントローラの他方側の前記ライトキャッシュおよび前記ビットマップに送信してライトコマンドを受信する工程と、前記コントローラの一方がシーズしているセグメントにヒットした場合に前記２つのコントローラの他方にシーズコマンドを送信することなく、前記ホストコンピュータからライトデータを受信して自己の前記ライトキャッシュに格納し当該格納位置に対応した自己の前記ビットマップを有効状態に更新する工程と、自己の前記ライトキャッシュのデータおよび自己の前記ビットマップを前記２つのコントローラの他方のＤＭＡコントローラを介して前記２つのコントローラの他方の前記ライトキャッシュと前記ビットマップに送信する工程を有することを特徴とする請求項１３に記載のキャッシュメモリ制御方法に存する。

【００２２】

【発明の実施の形態】本発明は、２コントローラ構成で、かつ、共有メモリのハードウェアを持たないディスクアレイ装置５０において、ライトデータを障害から守るための２重書きに関して、メイトコントローラ（換言すれば、相手コントローラ）のプロセッサに負荷をかけないことを特徴としている。

【００２３】ライトデータはコントローラ内のメモリに格納されるため、１コントローラ構成でライトキャッシュを使用すると、コントローラの故障によってライトデータを失ってしまう。一方、２コントローラ構成で、かつ、ライトデータを両方のコントローラに持っておけば、片方のコントローラが故障しても、ライトデータが失われることがない。このようにライトデータを両方のコントローラに保持する仕組みを２重書きと呼ぶ。以下、本発明の実施の形態を図面に基づいて詳細に説明する。

【００２４】（第１の実施の形態）図１は、本発明の第１の実施の形態に係るディスクアレイ装置５０を説明するための機能ブロック図である。本実施の形態のディスクアレイ装置５０は、ホストコンピュータ１１とディスク１５，…，１８との間に２つのコントローラ２１，３１を有するものである。

【００２５】ホストコンピュータ１１とコントローラ２１，３１は、例えば、ＳＣＳＩ（スカジー：スモール・コンピュータ・システム・インタフェース）バスのような汎用バス１２，１３で接続されている。

【００２６】コントローラ２１，３１とディスク１５，…，１８は、例えば、ＳＣＳＩバスのような汎用バス１４で接続されている。

【００２７】コントローラ２１は、マイクロプロセッサ２３と、リードまたはライトのデータをキャッシングしかつプロセッサの制御情報を記憶するメモリ２７と、ホストコンピュータ１１との通信を制御するインタフェースチップ２２と、ディスク１５，…，１８との通信を制御するインタフェースチップ２６と、自身のコントローラ２１のメモリ２７の任意アドレスのデータをメイトコントローラのメモリ２７の任意アドレスに送信できるＤＭＡコントローラ２４を有し、コントローラ２１内の各

チップは内部バス２５で接続されている。

【００２８】同様に、コントローラ３１は、マイクロプロセッサ３３と、リードまたはライトのデータをキャッシングしかつプロセッサの制御情報を記憶するメモリ３７と、ホストコンピュータ１１との通信を制御するインタフェースチップ３２と、ディスク１５，…，１８との通信を制御するインタフェースチップ３６と、自身のコントローラ３１のメモリ３７の任意アドレスのデータをメイトコントローラのメモリ３７の任意アドレスに送信できるＤＭＡコントローラ３４を有し、コントローラ３１内の各チップは内部バス３５で接続されている。

【００２９】図２は、図１のメモリ２７，３７に記憶する情報の定義図である。本発明ではセグメントと呼ばれる単位でキャッシュを管理しており、メモリ２７，３７には、セグメント管理情報４１，４４と、ビットマップ４２，４５と、ライトキャッシュ４３，４６が定義されている。セグメント管理情報４１，４４は、各セグメントの状態を保持するもので、シーズの状態を示すシーズフラグと、ＬＵＮ（ロジカルユニット番号）と、ＬＢＡ（ロジカルブロックアドレス）と、自コントローラのビットマップ格納位置を示す自ビットマップアドレスと、自コントローラのキャッシュ位置を示す自キャッシュアドレスと、メイトコントローラのビットマップ格納位置を示す他ビットマップアドレスと、メイトコントローラのキャッシュ位置を示す他キャッシュアドレスを有する。ビットマップ４２，４５は、各セグメントに対応するキャッシュのどの位置に有効なデータが存在するかを示すマップで、０が無効、１が有効である。ライトキャッシュ４３，４６は、各セグメントに対応するキャッシュメモリ自体である。

【００３０】コントローラ２１がホストコンピュータ１１からライトコマンドを受信すると、セグメント管理情報４１を参照して、自身がシーズしているセグメントにヒットしているかを確認し、ヒットしていなければ汎用バス１４を介してコントローラ３１にシーズコマンドを送信して、セグメントをシーズする。その後、ホストコンピュータ１１からライトデータを受信してライトキャッシュ４３に格納し、当該格納位置に対応したビットマップ４２を１に更新する。その後、ライトキャッシュ４３のデータとビットマップ４２をＤＭＡコントローラ２４，３４を介してコントローラ３１のメモリ３７のライトキャッシュ４６とビットマップ４５に送信する。ここまでの処理では、シーズコマンドの送受信によって、コントローラ３１のマイクロプロセッサ３３が動作している。

【００３１】続けて、ライトコマンドを受信し、自身がシーズしているセグメントにヒットした場合、コントローラ３１にシーズコマンドを送信することなく、ホストコンピュータ１１からライトデータを受信してライトキャッシュ４３に格納し、当該格納位置に対応したビットマップ４２を１に更新する。その後、ライトキャッシュ４３のデータとビットマップ４２をＤＭＡコントローラ２４，３４を介してコントローラ３１のライトキャッシュ４６とビットマップ４５に送信する。

【００３２】以上説明したように第１の実施の形態によれば、２回目以降のライトコマンドではコントローラ３１のプロセッサ３３に負荷をかけることなくライトコマンド処理が実行できるため、装置全体の性能が向上する。

【００３３】次にディスクアレイ装置５０の動作（キャッシュメモリ制御方法）について説明する。なお、本実施の形態の動作について、ホストコンピュータ１１からライトコマンドを受信し、それを２重書きする例を示す。また、本実施の形態では２回目以降のライトコマンド処理に効果が表れるため、ライトコマンドを連続して２回受信する場合で説明する。

【００３４】図３は、本発明の第１の実施の形態に係るキャッシュメモリ制御方法におけるライト処理を説明するためのフローチャートであって、汎用バス１４にＳＣＳＩインタフェースを使用し、ライトコマンドを２回受信した場合の処理の流れを示している。

【００３５】コントローラ２１は、ホストコンピュータ１１からライトコマンドを受信し（ステップＳ５１：ライトコマンド受信）、自身のセグメント管理情報４１を参照してセグメントのヒットミスの判定を行う（ステップＳ５２：ヒットミス判定）。１回目のコマンドなのでそれがミスし、コントローラ３１に汎用バス１４を介してシーズコマンド発行し（ステップＳ５３：ＳＣＳＩでシーズコマンド発行）、同時にそのパラメータとしてＬＵＮとＬＢＡを送信する。

【００３６】それを受信したコントローラ３１は、自身の空きセグメントをシーズして、セグメント管理情報４４に、シーズ状態を示すフラグとＬＵＮとＬＢＡとシーズしたキャッシュのアドレスとそれに対応するビットマップ４２，４５を格納するアドレスを登録して（ステップＳ５４：セグメント管理情報４４に登録）、コントローラ２１にシーズの完了報告を実行し（ステップＳ５５：ＳＣＳＩでシーズ完了報告）、同時にそのレスポンス情報としてキャッシュアドレスとビットマップ格納アドレスを送信する。コントローラ２１は自身のセグメント管理情報４１に、シーズ状態を示すフラグとＬＵＮとＬＢＡと自身がシーズしたキャッシュのアドレスとそれに対応するビットマップ４２，４５を格納するアドレスとメイトコントローラでシーズしたキャッシュのアドレスとそれに対応するビットマップ４２，４５を格納するアドレスを登録する（ステップＳ５６：セグメント管理情報４４に登録）。

【００３７】その後、ホストコンピュータ１１から自身がシーズしているキャッシュアドレスにライトデータを受信し（ステップＳ５７：ライトデータ受信）、それに

対応するビットマップ４２を更新する（ステップＳ５８：ビットマップ更新）。

【００３８】同ライトデータを、ＤＭＡコントローラ２４，３４を介してコントローラ３１がシーズしているキャッシュアドレスにダイレクトに転送し（ステップＳ５９：ＤＭＡでライトデータ転送）、同ビットマップ４２，４５を、同じくＤＭＡコントローラ２４，３４を介してコントローラ３１のビットマップ４５に転送し（ステップＳ６０：ＤＭＡでビットマップ転送）、これによって２重書きが完了し同時に１回目のライトコマンドが完了する。

【００３９】続けて、２回目のライトコマンドを受信すると（ステップＳ６１：ライトコマンド受信）、自身のセグメント管理情報４１を参照してセグメントのヒットミスの判定を行い（ステップＳ６２：ヒットミス判定）、それがヒットすると、１回目のようなシーズのやり取り無しにホストコンピュータ１１から自身がシーズしているキャッシュアドレスにライトデータを受信することができる（ステップＳ６３：ライトデータ受信）。

【００４０】その後、それに対応するビットマップ４２を更新して（ステップＳ６４：ビットマップ更新）、同ライトデータと同ビットマップ４２，４５を上記１回目と同様にしてコントローラ３１に転送し（ステップＳ６５：ＤＭＡでライトデータ転送、ステップＳ６６：ＤＭＡでビットマップ転送）、これによって２重書きが完了し同時に２回目のライトコマンドが完了する。

【００４１】これらの処理を比較すると、１回目のコマンド処理（ステップＳ６７）では、シーズコマンドのやり取りが行われて処理に時間がかかっているのに対し、２回目以降のコマンド処理（ステップＳ６８）ではシーズコマンドのやり取りがなく、処理時間が短縮され、かつ、コントローラ３１のプロセッサに負荷がかかっていないことがわかる。

【００４２】図４は、本発明の第１の実施の形態に係るディスクアレイ装置５０の各コントローラ２１，３１におけるライト処理を説明するためのフローチャートであって、コントローラ２１とコントローラ３１のライトコマンド処理のフローチャートを示している。

【００４３】図４を参照してそれぞれのプロセッサがどのような処理を行うかを説明する。ステップＳ７１（コントローラ２１の処理）の実行後、コントローラ２１は、ホストコンピュータ１１からライトコマンドを受信すると（ステップＳ７２：ホストコンピュータ１１からライトコマンド受信）、自身がシーズしているセグメントにヒットしているかどうかを判断する（ステップＳ７３：シーズ中セグメントにヒット？）。

【００４４】ここで１回目ならミスするので（ステップＳ７３のＮｏ）、ＳＣＳＩにてコントローラ３１へセグメントシーズコマンドを発行する（ステップＳ７４：ＳＣＳＩにてコントローラ３１へセグメントシーズコマンド発行）。

【００４５】コントローラ３１はそのシーズコマンドを受信（ステップＳ８２：コントローラ３１の処理→ステップＳ８３：ＳＣＳＩにてコントローラ２１からセグメントシーズコマンド受信）し、自身のセグメント管理情報４４を更新して（ステップＳ８４：セグメント管理情報４４の更新）、コントローラ２１にセグメントシーズ成功の通知を送信する（ステップＳ８５：ＳＣＳＩにてコントローラ２１へセグメントシーズ成功の通知を送信）。

【００４６】コントローラ２１はその通知を受信し（ステップＳ７５：ＳＣＳＩにてコントローラ３１からセグメントシーズ成功の通知を受信）、自身のセグメント管理情報４１を更新して（ステップＳ７６：セグメント管理情報４１の更新）、ホストコンピュータ１１からライトデータを受信し（ステップＳ７７：ホストコンピュータ１１からライトデータ受信）、それに対応してビットマップ４２を更新する（ステップＳ７８：ビットマップ４２の更新）。

【００４７】その後、ＤＭＡにてコントローラ３１にライトデータを送信し（ステップＳ７９：ＤＭＡにてコントローラ３１へライトデータを送信）、続いてビットマップ４２を送信し（ステップＳ８０：ＤＭＡにてコントローラ３１へビットマップ４２を送信）、これらの完了をもってホストコンピュータ１１へ完了報告を行う（ステップＳ８１：ホストコンピュータ１１へ完了報告）。

【００４８】２回目以降は、シーズ中セグメントのヒット判定（ステップＳ７３：シーズ中セグメントにヒット？）においてヒットとなり（ステップＳ７３のＹｅｓ）、シーズ関連の処理（ステップＳ７４：ＳＣＳＩにてコントローラ３１へセグメントシーズコマンド発行、ステップＳ７５：ＳＣＳＩにてコントローラ３１からセグメントシーズ成功の通知を受信、ステップＳ７６：セグメント管理情報４１の更新）をスキップする。これにより、コントローラ３１の処理（ステップＳ８２）が行われなくなるので、マイクロプロセッサ３３の負荷がかからない。

【００４９】以上説明したように第１の実施の形態によれば、メイトコントローラのセグメントをシーズしてそのキャッシュアドレスとビットマップ４２，４５を保持することにより、２回目以降のライトコマンド処理においてはメイトコントローラのプロセッサに負荷をかけることなく実行できるようになり、その結果、装置全体の性能の向上を図ることができるようになるといった効果を奏する。

【００５０】（第２の実施の形態）以下、本発明の第２の実施の形態を説明する。なお、上記第１の実施の形態において既に記述したものと同一の部分については、同一符号を付し、重複した説明は省略する。

【００５１】本発明の第２の実施の形態は、上記第１の

実施の形態の基本構成に加えて、コントローラ２１，３１数を増やすこともできる構成を備えている点に特徴を有している。具体的には、シーズコマンドを送信するための、例えば、ＳＣＳＩバスのような共用バスと、自身のメモリ２７，３７のデータを任意のコントローラ２１，３１のメモリ２７，３７にダイレクトに送信できるＤＭＡコントローラ（本図示）を有している。

【００５２】処理の流れは、上記第１の実施の形態の２コントローラ構成の場合と基本的に同じであり、２重書きしようとするコントローラ２１，３１に対して１回目はシーズコマンドを送信してセグメントをシーズし、２回目以降はシーズ処理無しで２重書きを実行する。

【００５３】なお、本発明が上記各実施の形態に限定されず、本発明の技術思想の範囲内において、各実施の形態は適宜変更され得ることは明らかである。また上記構成部材の数、位置、形状等は上記実施の形態に限定されず、本発明を実施する上で好適な数、位置、形状等にすることができる。また、各図において、同一構成要素には同一符号を付している。

【００５４】

【発明の効果】本発明は以上のように構成されているので、メイトコントローラのセグメントをシーズしてそのキャッシュアドレスとビットマップを保持することにより、２回目以降のライトコマンド処理においてはメイトコントローラのプロセッサに負荷をかけることなく実行できるようになり、その結果、装置全体の性能の向上を図ることができるようになるといった効果を奏する。

【図面の簡単な説明】

【図１】本発明の第１の実施の形態に係るディスクアレイ装置を説明するための機能ブロック図である。

【図２】図１のメモリに記憶する情報の定義図である。

【図３】本発明の第１の実施の形態に係るキャッシュメモリ制御方法におけるライト処理を説明するためのフローチャートである。

【図４】本発明の第１の実施の形態に係るディスクアレイ装置の各コントローラにおけるライト処理を説明するためのフローチャートである。

【図５】第１従来技術のキャッシュメモリ制御方式を説明するための機能ブロック図である。

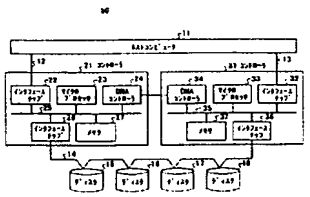
【図６】第２従来技術の主記憶装置の制御方式を説明するための機能ブロック図である。

【図７】第３従来技術のディスクアレイ装置を説明するための機能ブロック図である。

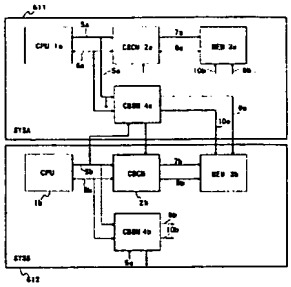
【符号の説明】

１１…ホストコンピュータ
１２，１３，１４…汎用バス
１５，…，１８…ディスク
２１，３１…コントローラ
２２，２６，３２，３６…インタフェースチップ
２３，３３…マイクロプロセッサ
２４，３４…ＤＭＡコントローラ
２５，３５…内部バス
２７，３７…メモリ
４１，４４…セグメント管理情報
４２，４５…ビットマップ
４３，４６…ライトキャッシュ
５０…ディスクアレイ装置

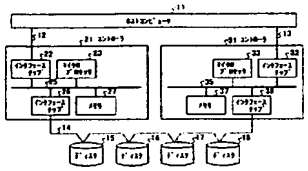
【図１】



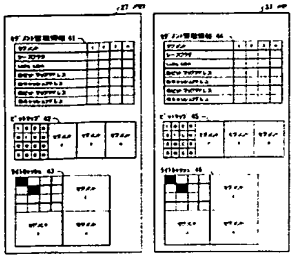
【図６】



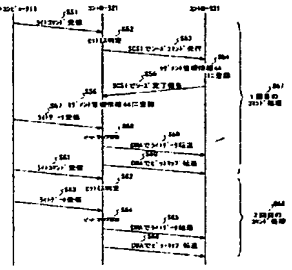
【図７】



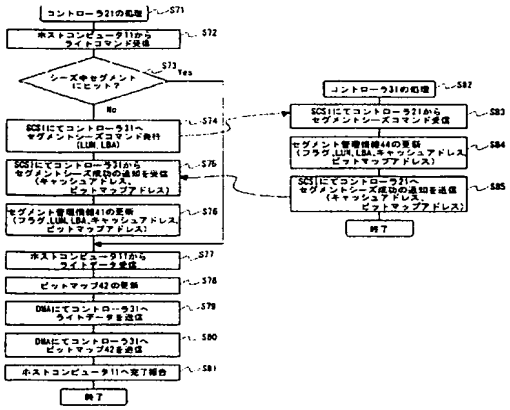
【図２】



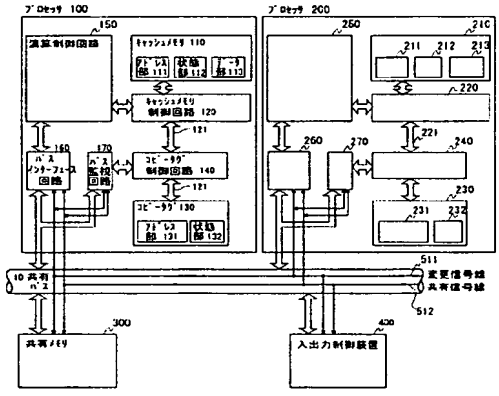
【図３】



【図４】

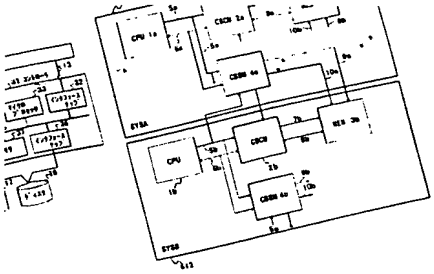


【図５】

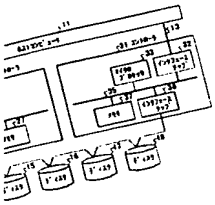


フロントページの続き

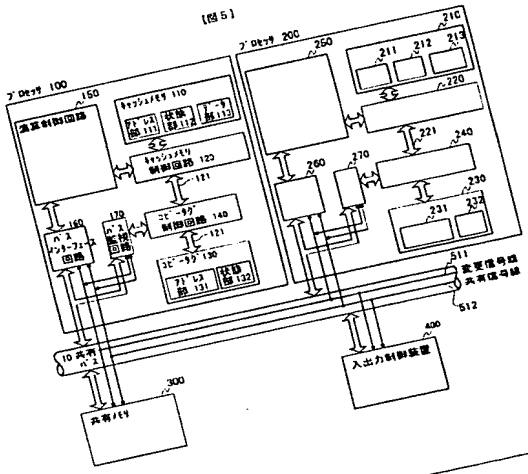
(51)Int.Cl.'	識別記号	FI	テーマコード（参考）
G06F 12/16	310	G06F 12/16	310J
	320		320L
G11B 19/02	501	G11B 19/02	501F



【図7】



【図5】



フロントページの続き

(51)Int.Cl.'	識別記号	F I	
G06F 12/16	310	G06F 12/16	310J
	320		320L
G11B 19/02	501	G11B 19/02	501F

Fターム(参考)

非打引文献 IBM Total Storage Enterprise Storage Server Model 800.
IBM Redbooks. SG24-6424-01. Second Edition (October 2002) IBM Corp.
ISBN 0738425256 pp58~59.

- Data that is accessed normally with some locality of reference will use partial track mode staging. This is the default mode.
- Data that is not a regular format, or where the history of access indicates that a full stage is required, will set the full track mode.
- The adaptive caching mode data is stored on disk and is reloaded at IML

Sequential reads

Cache space is released according to Least Recently Used (LRU) algorithms. Space in the cache used for sequential data is freed up quicker than other cache or record data. The ESS will continue to pre-stage sequential tracks when the last few tracks in a sequential staging group are accessed.

Stage requests for sequential operations can be performed in parallel on the RAID array, giving the ESS its high sequential throughput characteristic. Parallel operations can take place because the logical data tracks are striped across the physical data disks in the RAID array.

3.28 NVS and write operations

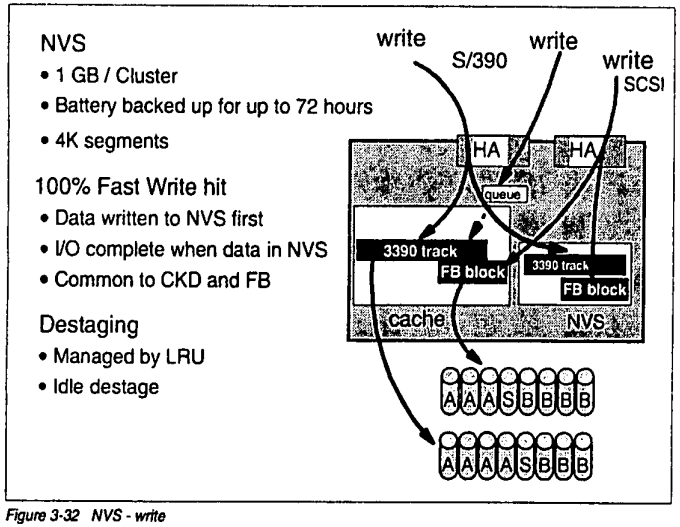


Figure 3-32 NVS - write

As Figure 3-32 illustrates, at any moment there are always two secured copies of any update into the ESS.

3.28.1 Write operations

Data written to an ESS is almost 100% fast write hits. A fast write hit occurs when the write I/O operation completes as soon as the data is in the ESS cache and non-volatile storage (NVS). The benefit of this is very fast write operations.

Fast write

Data received by the host adapter is transferred first to the NVS and a copy held in the host adapter buffer. The host is notified that the I/O operation is complete as soon as the data is in NVS. The host adapter, once the NVS transfer is complete, then transfers the data to the cache.

The data remains in the cache and NVS until it is destaged. Destage is triggered by cache and NVS usage thresholds.

3.28.2 NVS

The NVS size is 2 GB (1 GB per cluster). The NVS is protected by a battery. The battery will power the NVS for up to 72 hours following a total power failure.

NVS LRU

NVS is managed by a Least Recently Used (LRU) algorithm. The ESS attempts to keep free space in the NVS by anticipatory destaging of tracks when the space used in NVS exceeds a threshold. In addition, if the ESS is idle for a period of time, an idle destage function will destage tracks until, after about 5 minutes, all tracks will be destaged.

Both cache and NVS operate on LRU lists. Typically space in the cache occupied by sequential data is released earlier than space occupied by data that is likely to be re-referenced. Sequential data in the NVS is destaged ahead of random data.

When destaging tracks, the ESS attempts to destage all the tracks that would make up a RAID stripe, minimizing the RAID-related activities in the SSA adapter.

NVS location

NVS for cluster 1 is located physically in I/O drawer of cluster 2, and vice versa. This ensures that we always have one good copy of data, should we have a failure in one cluster.

See 3.8, "Cluster operation: failover/failback" on page 60 for more information.